Programming Assignment 3: SDN and Ryu

November 5, 2023

Instruction: After running ryu controller, run mininet with the command: `sudo mn --custom topo.py --topo mytopo --controller remote,ip=127.0.0.1,port=6633 --mac`

1 Controller Hub and Learning Switch

Controller Hub

h2 ping h5

- **Ping 1:** 6 packets transmitted, 6 received, 0% packet loss, time 5007ms. rtt min/avg/max/mdev = 7.741/10.723/19.279/4.099 ms
- **Ping 2:** 6 packets transmitted, 6 received, 0% packet loss, time 5007ms. rtt min/avg/max/mdev = 4.430/6.717/8.415/1.653 ms
- **Ping 3:** 5 packets transmitted, 5 received, 0% packet loss, time 4007ms. rtt min/avg/max/mdev = 4.433/6.505/9.608/2.355 ms

Throughput Test: iperf h1 h5

['9.91 Mbits/sec', '11.5 Mbits/sec']

Installed Rules after pingall

```
mininet> sh ovs-ofctl dump-flows s1
 cookie=0x0, duration=17.662s, table=0, n_packets=164, n_bytes=14717, priority=0 actions=CONTROLLER:65535
mininet> sh ovs-ofctl dump-flows s2
 cookie=0x0, duration=19.155s, table=0, n_packets=165, n_bytes=14803, priority=0 actions=CONTROLLER:65535
mininet>
```

Learning Switch

h2 ping h5

- **Ping 1:** 37 packets transmitted, 37 received, 0% packet loss, time 36826ms. rtt min/avg/max/mdev = 0.109/0.875/25.926/4.175 ms
- **Ping 2:** 7 packets transmitted, 7 received, 0% packet loss, time 6113ms. rtt min/avg/max/mdev = 0.130/0.319/1.180/0.355 ms
- **Ping 3:** 6 packets transmitted, 6 received, 0% packet loss, time 5104ms. rtt min/avg/max/mdev = 0.160/0.387/1.361/0.435 ms

Throughput Test: iperf h1 h5

['10.7 Gbits/sec', '10.7 Gbits/sec']

Installed Rules after pingall

```
mininet> sh ovs-ofctl dump-flows s1
 cookie=0x0, duration=39.302s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth2",dl_dst=00:00:00:00:00:01 actions=output:"s1-eth1"
 cookie=0x0, duration=39.299s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_dst=00:00:00:00:00:02 actions=output:"s1-eth2"
 cookie=0x0, duration=39.287s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth3",dl_dst=00:00:00:00:00:01 actions=output:"s1-eth1"
 cookie=0x0, duration=39.284s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_dst=00:00:00:00:00:03 actions=output:"s1-eth3"
 cookie=0x0, duration=39.268s, table=0, n_packets=7, n_bytes=518, priority=1,in_port="s1-eth4",dl_dst=00:00:00:00:00:01 actions=output:"s1-eth1"
 cookie=0x0, duration=39.266s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_dst=00:00:00:00:00:04 actions=output:"s1-eth4"
 cookie=0x0, duration=39.245s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_dst=00:00:00:00:00:05 actions=output:"s1-eth4"
 cookie=0x0, duration=39.223s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth3",dl_dst=00:00:00:00:00:02 actions=output:"s1-eth2"
 cookie=0x0, duration=39.220s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth2",dl_dst=00:00:00:00:00:03 actions=output:"s1-eth3"
 cookie=0x0, duration=39.203s, table=0, n_packets=149522, n_bytes=9868548, priority=1,in_port="s1-eth4",dl_dst=00:00:00:00:00:02 actions=output:"s1-eth2"
 cookie=0x0, duration=39.198s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth2",dl_dst=00:00:00:00:00:04 actions=output:"s1-eth4"
 cookie=0x0, duration=39.179s, table=0, n_packets=152391, n_bytes=6677428350, priority=1,in_port="s1-eth2",dl_dst=00:00:00:00:00:05 actions=output:"s1-eth4"
 cookie=0x0, duration=39.152s, table=0, n_packets=7, n_bytes=518, priority=1,in_port="s1-eth4",dl_dst=00:00:00:00:00:03 actions=output:"s1-eth3"
 cookie=0x0, duration=39.147s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth3",dl_dst=00:00:00:00:00:04 actions=output:"s1-eth4"
 cookie=0x0, duration=39.130s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth3",dl_dst=00:00:00:00:00:05 actions=output:"s1-eth4"
 cookie=0x0, duration=41.792s, table=0, n_packets=118, n_bytes=12304, priority=0 actions=CONTROLLER:65535
mininet> sh ovs-ofctl dump-flows s2
 cookie=0x0, duration=41.376s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth1",dl_dst=00:00:00:00:00:01 actions=output:"s2-eth3"
 cookie=0x0, duration=41.368s, table=0, n_packets=8, n_bytes=616, priority=1,in_port="s2-eth3",dl_dst=00:00:00:00:00:04 actions=output:"s2-eth1"
 cookie=0x0, duration=41.355s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_dst=00:00:00:00:00:01 actions=output:"s2-eth3"
 cookie=0x0, duration=41.346s, table=0, n_packets=152397, n_bytes=6677428826, priority=1,in_port="s2-eth3",dl_dst=00:00:00:00:00:05 actions=output:"s2-eth2"
 cookie=0x0, duration=41.311s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth1",dl_dst=00:00:00:00:00:02 actions=output:"s2-eth3"
 cookie=0x0, duration=41.290s, table=0, n_packets=149518, n_bytes=9868268, priority=1,in_port="s2-eth2",dl_dst=00:00:00:00:00:02 actions=output:"s2-eth3"
 cookie=0x0, duration=41.260s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth1",dl_dst=00:00:00:00:00:03 actions=output:"s2-eth3"
 cookie=0x0, duration=41.239s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_dst=00:00:00:00:00:03 actions=output:"s2-eth3"
 cookie=0x0, duration=41.207s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_dst=00:00:00:00:00:04 actions=output:"s2-eth1"
 cookie=0x0, duration=41.203s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s2-eth1",dl_dst=00:00:00:00:00:05 actions=output:"s2-eth2"
 cookie=0x0, duration=43.892s, table=0, n_packets=113, n_bytes=11758, priority=0 actions=CONTROLLER:65535
mininet>
```

Comparison between Learning Switch and Hub Controller

- The latency of Hub Controller is more compared to Learning Switch because Learning Switch consults the controller for the first packet to a unknown destination and installs rules to avoid further consultation with the controller, this makes Latency of Learning Switch low. Hub consults the controller for every packet and does not install any rule, thus latency is high.
- For the same reason, throughput is high in case of Learning Switch.

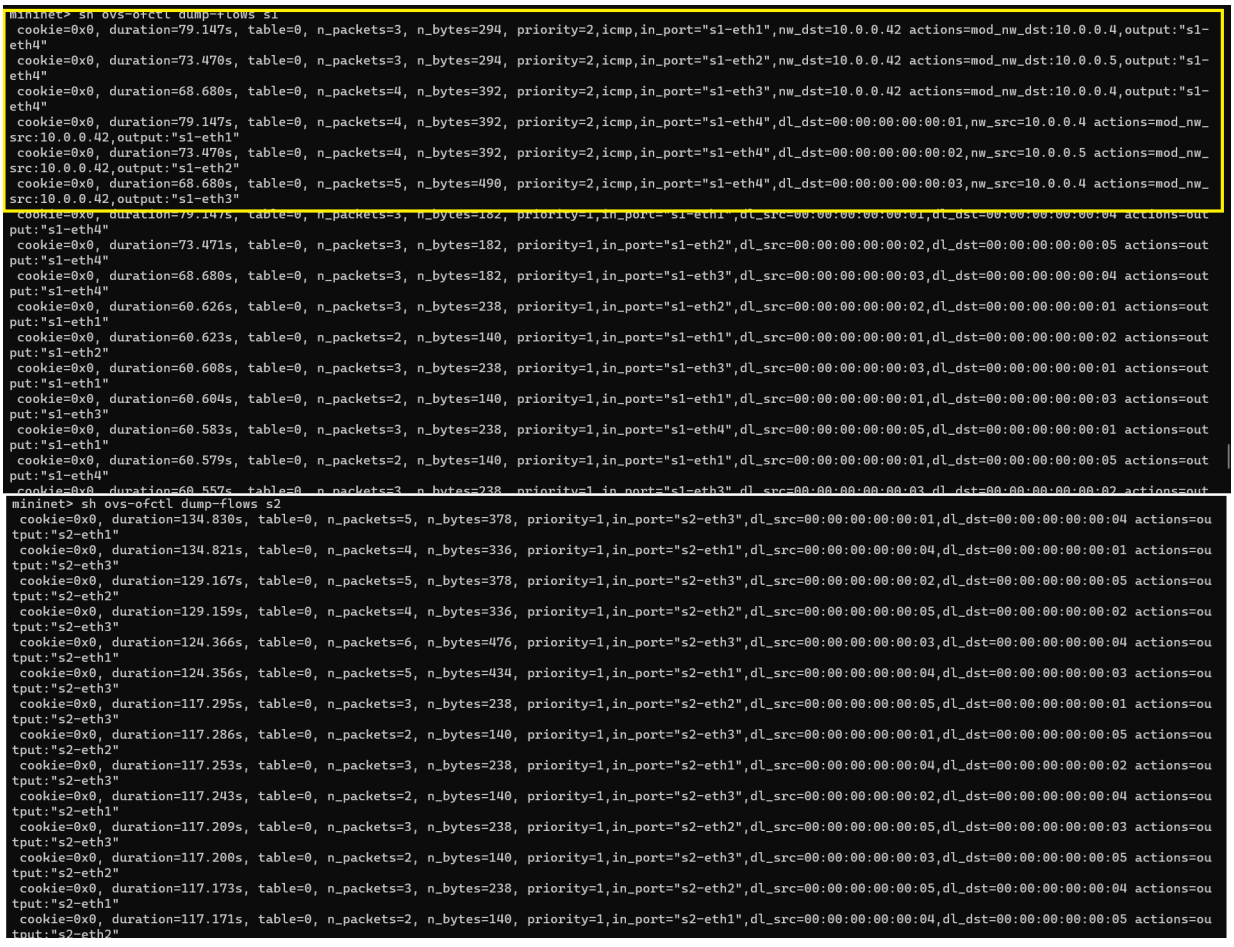
2 Firewall and Monitor

Note: Since, nothing is mentioned about where to install firewall rules, I've installed them both on S1 and S2. However, installing on only one switch will also work for the given topology. The monitor functionality is implemented only on S1.

Pingall Results

```
h1 -> h2 h3 X h5
h2 -> h1 h3 h4 X
h3 -> h1 h2 h4 X
h4 -> X h2 h3 h5
h5 -> h1 X X h4
```

Installed Rules



Minimizing Rules and handling conflicting rules

- Firstly, firewall rules are needed in only one switch. As we can see, there are low priority rules to allow communication between two hosts and then there are high priority rules to block communication between them. We can selectively allow instead of first allowing all, then blocking, this will minimize the number of installed rules. Also, we can use wildcard matching to reduce the number of installed rules.
- To ensure that pre-existing rules do not interfere with the newly added rules, the network operator has to use the priority of rules properly.

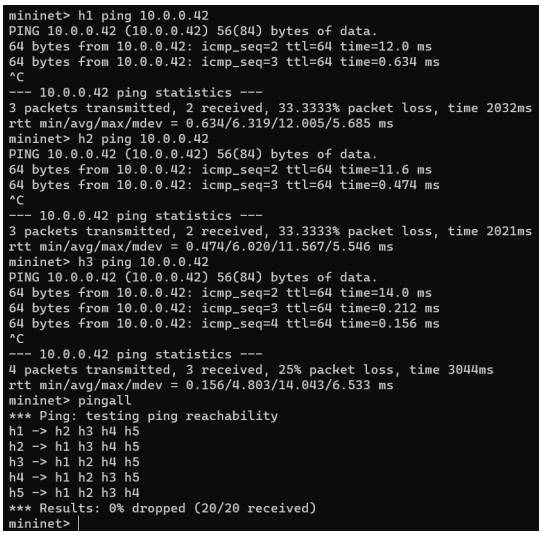
3 Load Balancer

Installed Rules

```
mininet> sh ovs-ofctl dump-flows s1
 cookie=0x0, duration=79.147s, table=0, n_packets=3, n_bytes=294, priority=2,icmp,in_port="s1-eth1",nw_dst=10.0.0.42 actions=mod_nw_dst:10.0.0.4,output:"s1-
eth4"
 cookie=0x0, duration=73.470s, table=0, n_packets=3, n_bytes=294, priority=2,icmp,in_port="s1-eth2",nw_dst=10.0.0.42 actions=mod_nw_dst:10.0.0.5,output:"s1-
eth4"
 cookie=0x0, duration=68.680s, table=0, n_packets=4, n_bytes=392, priority=2,icmp,in_port="s1-eth3",nw_dst=10.0.0.42 actions=mod_nw_dst:10.0.0.4,output:"s1-
eth4"
 cookie=0x0, duration=79.147s, table=0, n_packets=4, n_bytes=392, priority=2,icmp,in_port="s1-eth4",dl_dst=00:00:00:00:00:01,nw_src=10.0.0.4 actions=mod_nw_
src:10.0.0.42,output:"s1-eth1"
 cookie=0x0, duration=73.470s, table=0, n_packets=4, n_bytes=392, priority=2,icmp,in_port="s1-eth4",dl_dst=00:00:00:00:00:02,nw_src=10.0.0.5 actions=mod_nw_
src:10.0.0.42,output:"s1-eth2"
 cookie=0x0, duration=68.680s, table=0, n_packets=5, n_bytes=490, priority=2,icmp,in_port="s1-eth4",dl_dst=00:00:00:00:00:03,nw_src=10.0.0.4 actions=mod_nw_
src:10.0.0.42,output:"s1-eth3"
 cookie=0x0, duration=79.147s, table=0, n_packets=3, n_bytes=182, priority=1,in_port="s1-eth1",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:04 actions=out
put:"s1-eth4"
 cookie=0x0, duration=73.471s, table=0, n_packets=3, n_bytes=182, priority=1,in_port="s1-eth2",dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:05 actions=out
put:"s1-eth4"
 cookie=0x0, duration=68.680s, table=0, n_packets=3, n_bytes=182, priority=1,in_port="s1-eth3",dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:04 actions=out
put:"s1-eth4"
 cookie=0x0, duration=60.626s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth2",dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:01 actions=out
put:"s1-eth1"
 cookie=0x0, duration=60.623s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:02 actions=out
put:"s1-eth2"
 cookie=0x0, duration=60.608s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth3",dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:01 actions=out
put:"s1-eth1"
 cookie=0x0, duration=60.604s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:03 actions=out
put:"s1-eth3"
 cookie=0x0, duration=60.583s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth4",dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:01 actions=out
put:"s1-eth1"
 cookie=0x0, duration=60.579s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s1-eth1",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:05 actions=out
put:"s1-eth4"
 cookie=0x0, duration=60.557s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s1-eth3",dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:02 actions=out
```

```
mininet> sh ovs-ofctl dump-flows s2
 cookie=0x0, duration=134.830s, table=0, n_packets=5, n_bytes=378, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:04 actions=ou
tput:"s2-eth1"
 cookie=0x0, duration=134.821s, table=0, n_packets=4, n_bytes=336, priority=1,in_port="s2-eth1",dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:01 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=129.167s, table=0, n_packets=5, n_bytes=378, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:05 actions=ou
tput:"s2-eth2"
 cookie=0x0, duration=129.159s, table=0, n_packets=4, n_bytes=336, priority=1,in_port="s2-eth2",dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:02 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=124.366s, table=0, n_packets=6, n_bytes=476, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:04 actions=ou
tput:"s2-eth1"
 cookie=0x0, duration=124.356s, table=0, n_packets=5, n_bytes=434, priority=1,in_port="s2-eth1",dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:03 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=117.295s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:01 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=117.286s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:05 actions=ou
tput:"s2-eth2"
 cookie=0x0, duration=117.253s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth1",dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:02 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=117.243s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:04 actions=ou
tput:"s2-eth1"
 cookie=0x0, duration=117.209s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:03 actions=ou
tput:"s2-eth3"
 cookie=0x0, duration=117.200s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s2-eth3",dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:05 actions=ou
tput:"s2-eth2"
 cookie=0x0, duration=117.173s, table=0, n_packets=3, n_bytes=238, priority=1,in_port="s2-eth2",dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:04 actions=ou
tput:"s2-eth1"
 cookie=0x0, duration=117.171s, table=0, n_packets=2, n_bytes=140, priority=1,in_port="s2-eth1",dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:05 actions=ou
tput:"s2-eth2"
```

Ping to 10.0.0.42



Implementing Load Balancing Policy

To implement load balancing policy based on server load, the controller can implement traffic monitoring to get the flow stats for the servers which can be used as a measure of how loaded a server is and based on that it can do load balancing.